# ЛАБОРАТОРНАЯ РАБОТА № 13

**Делегаты как свойства**

**Цель работы:** Ознакомиться с использованием делегатов в качестве свойств класса.

**Вариант 1**: Написать программу, которая вводит с файла/клавиатуры данные и выдает их на экран, сортируя по одному из трех признаков, указанному оператором. Данные содержат информацию о: Студенте, его успеваемости.

**Ход работы:**

static void Main(string[] args)

{

List<Student> students = new List<Student>();

// Считываем данные из файла или с клавиатуры

Console.WriteLine("Выберите источник данных (файл/клавиатура):");

string source = Console.ReadLine().ToLower();

if (source == "файл")

{

Console.WriteLine("Введите путь к файлу:");

string filePath = Console.ReadLine();

ReadFromFile(filePath, students);

}

else if (source == "клавиатура")

{

ReadFromKeyboard(students);

}

else

{

Console.WriteLine("Некорректный источник данных.");

return;

}

// Выбор критерия сортировки

Console.WriteLine("Выберите критерий сортировки (имя/возраст/средний балл):");

string sortBy = Console.ReadLine().ToLower();

// Сортировка и вывод результатов

SortAndPrint(students, sortBy);

Console.ReadKey();

}

static void ReadFromFile(string filePath, List<Student> students)

{

try

{

string[] lines = File.ReadAllLines(filePath);

foreach (var line in lines)

{

string[] parts = line.Split(',');

students.Add(new Student

{

Name = parts[0],

Age = int.Parse(parts[1]),

GradePointAverage = double.Parse(parts[2])

});

}

}

catch (Exception e)

{

Console.WriteLine($"Ошибка при чтении файла: {e.Message}");

}

}

static void ReadFromKeyboard(List<Student> students)

{

Console.WriteLine("Введите данные о студентах (имя, возраст, средний балл). Для завершения введите 'конец':");

string input;

while ((input = Console.ReadLine()) != "конец")

{

string[] parts = input.Split(',');

students.Add(new Student

{

Name = parts[0],

Age = int.Parse(parts[1]),

GradePointAverage = double.Parse(parts[2])

});

}

}

static void SortAndPrint(List<Student> students, string sortBy)

{

switch (sortBy)

{

case "имя":

students.Sort((x, y) => string.Compare(x.Name, y.Name));

break;

case "возраст":

students.Sort((x, y) => x.Age.CompareTo(y.Age));

break;

case "средний балл":

students.Sort((x, y) => x.GradePointAverage.CompareTo(y.GradePointAverage));

break;

default:

Console.WriteLine("Некорректный критерий сортировки.");

return;

}

Console.WriteLine("Результаты сортировки:");

foreach (var student in students)

{

Console.WriteLine($"Имя: {student.Name}, Возраст: {student.Age}, Средний балл: {student.GradePointAverage}");

}

}

class Student

{

public string Name { get; set; }

public int Age { get; set; }

public double GradePointAverage { get; set; }

}

**Вывод:** Ознакомился с использованием делегатов в качестве свойств класса.